`

Knowledge Base Article

Product Group: Software
Product: CMSW7700 - @ptitude Monitoring Suite
Version: N/A

Abstract

SKF @ptitude Monitoring Suite allows for the import and export of data through .MAB files. There are different ways of importing the data contained in these backup files. During an import, the application will look for existing elements in the database. Instead of overwriting an element, it will create the element and append a .1, .2, .3, etc. to the name. This is especially common with frequencies. This clutters the list of frequencies and is not always easy to clean up. This article gives instructions on how to remove these unwanted duplicate frequency entries.

Overview

The procedure described in this article is specific to SQL Server databases. Running the statements against an Oracle database may generate syntax errors.

- Review all of the steps in full to thoroughly understand what each one entails, prior to starting this procedure.
- Create a backup of the database before proceeding.
- Ensure that all users are logged out of @ptitude Analyst and that all SKF services are shut down (Transaction Server, IMx service, Microlog Inspector Service, SKF Monitor Recovery service, etc.).

1. Create a backup of the bearing table and name it “BearingBak” using the following statement:

```
-- Backup bearing table
Select * into BearingBak from Bearing
```

2. The following statement will look for a table **BearingDelete** (which should not exist) and drop it:

```
-- Drop and create temp table for duplicate bearings.
if exists (select * from dbo.sysobjects where id =
object_id(N'BearingDelete') and ObjectProperty(id, N'IsTable') = 1)
  Drop table BearingDelete
```

3. Use the following statement to capture only the duplicate frequencies and insert them into the BearingDelete table:

```
-- Use the select statement below to create BearingDelete table to hold
data to be deleted.
SELECT  bearingid, name, manufacture, SUBSTRING(name, 1,
CHARINDEX('.',name,1) - 1 ) [NewName]
INTO BearingDelete
FROM bearing
WHERE
  name  like '%.1' or
  name  like '%.2' or
  name  like '%.3' or
  name  like '%.4' or
  name  like '%.5' or
  name  like '%.6' or
  name  like '%.7' or
  name  like '%.8' or
  name  like '%.9'
order by bearingid
```

4. Now that the duplicate frequencies in the BearingDelete table are there, execute the delete statement to remove them from the database:

```
-- Delete duplicate bearings
Delete from bearing where bearingid in (Select bearingid from
BearingDelete)
```

5. Log in to @ptitude Analyst and confirm the duplicate frequencies no longer exist in the database.

`

6. Run the statement below to remove the temp table BearingDelete from the database:

```
-- Drop temp table.
if exists (select * from dbo.sysobjects where id =
object_id(N'BearingDelete') and ObjectProperty(id, N'IsTable') = 1)
  Drop table BearingDelete
```

7. Exit out of Microsoft SQL Server Management Studio.

8. If any issues arise, the TSG Team can assist with restoring the table. Restoring the backup is always an option as well.

*** THE PROCEDURE IS COMPLETE. THE INFORMATION BELOW IS ONLY TO BE USED IF THE BEARING TABLE NEEDS TO BE RESTORED. ***

```
-- To restore Bearing table if problem exists
Alter table bearing disable  trigger all
SET IDENTITY_INSERT BEARING ON
Truncate table Bearing

Insert into bearing (BearingId, Name, Manufacture, Description, BPFO,
BPFI, BSF, FTF, EDITSTATUS)
Select BearingId, Name, Manufacture, Description, BPFO, BPFI, BSF,
FTF, EDITSTATUS from BearingBak

SET IDENTITY_INSERT BEARING OFF
Alter table bearing enable  trigger all
```

For further assistance, please contact the Technical Support Group by phone at 1-800-523-7514 option 8, or by email at TSG-Americas@skf.com.